PORTAL
THE ACM DIGITAL LIBRARY

Try the *new* Portal design
Give us your opinion after using it.

Citation

Conference on Information and Knowledge Management

>archive

Proceedings of the sixth international conference on Information and knowledge management >toc

1997 , Las Vegas, Nevada, United States

Discovering similar resources by content part-linking

Authors
Brad Perry
Wesley W. Chu

Sponsors
SIGIR : ACM Special Interest Group on Information Retrieval
SIGLINK : Hypertext, Hypermedia, and Web
SIGWEB : ACM Special Interest Group on Hypertext, Hypermedia, and Web
SIGGROUP : ACM Special Interest Group on Supporting Group Work
SIGART : ACM Special Interest Group on Artificial Intelligence

Publisher
ACM Press New York, NY, USA

Pages: 317 - 324 Series-Proceeding-Article
Year of Publication: 1997
ISBN:0-89791-970-X

doi> http://doi.acm.org/10.1145/266714.266918 (Use this link to Bookmark this page)

> full text > references > index terms > peer to peer

> Discuss > Similar > Review this Article Save to Binder

> BibTex Format

↑ FULL TEXT: Access Rules

pdf 1.10 MB

↑ REFERENCES

Note: OCR errors may be found in this Reference List extracted from the full text article. ACM has opted to expose the complete List rather than only correct and linked references.

1 James Allan, Automatic hypertext construction, Cornell University, Ithaca, NY, 1996

2 James Allan, Automatic hypertext link typing, Proceedings of the the seventh ACM conference on Hypertext, p.42-52, March 16-20, 1996, Bethesda, Maryland, United States

3 AltaVista. Atta Vista Search: Main Page. http://www.altavista.di~tal.com[, 1995.

4 1L Armstrong, D. Freitag, T. Joac~ims, and T. Mitchell. Webwatcher: A learning apprentice for the world wide web. In 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, 1995.

5 Mark Bernstein, An apprentice that discovers hypertext links, Hypertext: concepts, systems and applications, Cambridge University Press, New York, NY, 1992

6 Vamtevax Bush. As we may think. Atlantic Monthly, July 1945.

7 Fazli Can, Incremental clustering for dynamic information processing, ACM Transactions on Information Systems (TOIS), v.11 n.2, p.143-164, April 1993

8 Chip Cleary , Ray Bareiss, Practical methods for automatically generating typed links, Proceedings of the the seventh ACM conference on Hypertext, p.31-41, March 16-20, 1996, Bethesda, Maryland, United States

9 M. Hearst. Multi-paxagraph segmentation of expository text. In A C[,-9~, 1994.

10 L. Page. BackRub WebCrawler. http://backrub.stanford.edu/, 1996.

11 John Bradley Perry, Dynamically discovering similar resources in large-scale information networks, University of California at Los Angeles, Los Angeles, CA, 1998

12 Airi Salminen , Jean Tague-Sutcliffe , Charles McClellan, From text to hypertext by indexing, ACM Transactions on Information Systems (TOIS), v.13 n.1, p.69-99, Jan. 1995

13 Gerard Salton, Automatic text processing: the transformation, analysis, and retrieval of information by computer, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 1989

14 W3C. Relatio~kips in ttTML links. Technical ReporL mp:/lwww.~3.o~glpublWWW/~a~kVp/Reh~ion~hips, World Wide Web Consortium, May 1996.

15 WSC. World Wide Web Consortium. http://www.w3.org/, 1996.

16 Ron Weiss , Bienvenido Vélez , Mark A. Sheldon, HyPursuit: a hierarchical network search engine that exploits content-link hypertext clustering, Proceedings of the the seventh ACM conference on Hypertext, p.180-193, March 16-20, 1996, Bethesda, Maryland, United States

17 T.W. Yen, M. Jacobsen, H. Garcia-Molina, and U. Dayal. From user access patterns to dynamic hypertext linking. Stanford UniversRy: Working Papers, 1996.

18 L. De Young. Links considered harmful In Hyperlext: Concepts, Systems, and Applicaffons: European Conference on Hypertext, 1990.

↑ INDEX TERMS

Primary Classificati n:

H. Information Systems

↪ **H.2** DATABASE MANAGEMENT

Additional Classificati n:

C. Computer Systems Organization

↪ **C.2** COMPUTER-COMMUNICATION NETWORKS

↪ **C.2.1** Network Architecture and Design

↪ **Nouns:** Internet

H. Information Systems

↪ **H.3** INFORMATION STORAGE AND RETRIEVAL

↪ **H.3.3** Information Search and Retrieval

↪ **Subjects:** Search process

↪ **H.5** INFORMATION INTERFACES AND PRESENTATION (I.7)

I. Computing Methodologies

↪ **I.6** SIMULATION AND MODELING

↪ **I.7** DOCUMENT AND TEXT PROCESSING

↪ **I.7.2** Document Preparation

↪ **Nouns:** HTML

General Terms:
Design, Experimentation, Languages, Management, Measurement, Performance, Theory

↑ **Peer to Peer - Readers of this Article have also read:**

- M^4: a metamodel for data preprocessing
  Proceedings of the 4th ACM international workshop on Data warehousing and OLAP
  Anca Vaduva , Jörg-Uwe Kietz , Regina Zücker

- Presenting computer algorithm knowledge units in computer science curriculum
  The Journal of Computing in Small Colleges 16, 2
  S. Krishnaprasad

- LR Parsing
  ACM Computing Surveys (CSUR) 6, 2
  A. V. Aho , S. C. Johnson

- Developing teamwork through experiential learning
  The Journal of Computing in Small Colleges 16, 2
  Courtney S. Ferguson , Shade K. Little , Marilyn K. McClelland

- A pilot study for developing a quantitative model for outcomes assessment in the computer science program at a Small University
  The J urnal of Computing in Small C lleges 16, 2
  Robert O. Jarman , Sankara N. Sethuraman

The ACM Portal is published by the Association for Computing Machinery. C pyright © 2004 ACM, Inc.

Discovering Similar Resources by Content Part-Linking *

Brad Perry[†] Wesley W. Chu
{bjperry, wwc}@cs.ucla.edu
Computer Science Department
University of California, Los Angeles

Abstract

This paper develops a novel approach to compute and search for context-specific similarity relationships between HTML resources (HyperText Markup Language). The similarity is computed by decomposing resources into parts, finding similar parts among resources, and then extracting the *pattern of matched parts* into a feature vector. This general approach for resource-resource linking is termed *part-linking* and its specialization to HTML resources is termed *html-linking*. Given the vectors describing resource-resource associations, the following search process can be employed: "Find all resources similar to r_s in the same manner that resource r_d is similar to r_s." A *neighborhood search model* is developed to execute and control this style of request and find all resources in the network matching the sample structure within well-defined windows of approximation. The innovation of the html-linking model is its matching of textual resources based on the locality and organizational structure of the content found to be matching between resources.

1 Introduction

The amount, variety, and distribution of online information is rapidly exploding with the advent of the World Wide Web (WWW) information space in the global Internet. Information resources change constantly, and users are faced with the daunting challenges of finding, navigating, collecting, evaluating, and processing in this dynamic information universe. *Intelligently integrating and correlating* information has surfaced as a breakthrough topic governing the success of next-generation information networks such as the WWW. This topic is concerned with scalable methods and metaphors to abstract, integrate, fuse, or otherwise reduce and add value to massive amounts of distributed information.

One mechanism for bringing such content-orientation to the web is to "link" information resources to other semantically related resources in the network. Such a value-added *link service* would monitor the information space and attempt to link resources when their internal content exhibits semantically important patterns.

This paper introduces a content linking service based on answering questions of the style, "Find all resources similar to r_s in the same manner that resource r_d is similar to r_s." The content association exhibited between r_s and r_d is taken as the *association type* to search for from resource r_s in the information network. We introduce the concept of *part-linking* to identify and evaluate the association between textual resources. The part-linking concept is then specialized to apply to HTML resources, resulting in the *html-linking* model for linking resources on the web. The innovation of html-linking lies in that it computes the association between resources based on the *locality* and *organizational structure* of the content matching between two resources.

1.1 Background and Motivation

The general mechanism of linking resources is what is commonly referred to as *hypertext* and has become a fundamental tool for navigating and organizing large-scale information collections [6, 18]. A pointer from one resource to another is referred to as a "hypertext link" (or just a "link"). A hypertext link is, at its most basic level, a connection between two units of text from two resources. Without any further information describing the link, it serves only as a casual suggestion that relevant information may exist at the other end of the link. A *link type* is an attribute associated with each hypertext link that gives some idea of the effect, or semantic intent, of following the link [18]. Typed links are essential for providing contextual focus to navigation in large document collections. This is becoming apparent in the web community, one of the largest testbeds for hypertext developments, where the WWW consortium is actively considering the class of *link types* that will be incorporated into the next versions of the HTML hypertext standard [15, 14]. Given the size and diversity of large-scale information networks, it has become important to explore techniques for automatically (or semi-automatically) implanting typed links over the information space [5, 1, 16, 7, 8, 12, 17, 4, 10]. Within the auspices of the WWW, these techniques fall into two general categories: link discovery and link annotation.

In *link discovery* methods [5, 1, 16, 7, 8, 12], network resources are manipulated as individual documents and the goal is to establish semantic links between semantically related documents. In this respect, the links are discovered and maintained "above" the document space in separate link

*This research is supported in part by DARPA contracts F30602-94-C-0207 and N66001-97-C-8601

[†] This author is now at MCC in Austin, TX; bperry@mcc.com

Permission to make digital/hard copies of all or part of this material for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication and its date appear, and notice is given that copyright is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires specific permission and/or fee.

CIKM 97 LasVegas Nevada USA
Copyright 1997 ACM 0-89791-970-x/97/11..$3.50

services.

The particular structure of the WWW and its dominant encoding standard, the Hypertext Markup Language (HTML), presents opportunities for some unique developments in resource linking services. HTML currently supports a single typeless link that authors can place inside their documents. The existence of such embedded associations in the information space lends itself to a new type of resource linking service – that of *annotating* the existing link base with labels of the semantic intent, or association, between the linked resources. In other words, HTML authors point to resources related to their current documents and the *link annotation* services [17, 4, 10] add types or context to these links with respect to how they are used in and across the information space. In this respect, the types associated with links are discovered and maintained "above" the typeless HTML information space in separate link annotation services.

In both the link discovery and link annotation approaches, the central issue is that of *how to compute and capture the semantic, or content-sensitive, association between resources. Html-linking* is the novel resource association technique introduced in this paper. Specifically, our html-linking model develops an innovative method for capturing the association between HTML resources in a manner sensitive to the *locality* and *organizational structure* of the content matching between two resources. As a result, the html-linking methods can be used as the resource comparison technique at the center of both link discovery and link annotation approaches for bringing content-orientation to the web.

2 Linking Resources by Part Patterns

Html-linking is developed by introducing a concept of resource *part-linking* and then specializing this concept to apply to HTML resources. The idea behind part-linking is the "part comparison" techniques originally identified in [1, 2]:

> For the analysis of resource-resource relationships, resources must be *broken into parts*. Depending on the nature of the collection a part may be a paragraph, sentence group, sentence, or other sequence of text. The set of part-pairs exhibiting "sufficient" similarity are analyzed and the general pattern defined by these pairs determines the semantic association assigned between the resources.

Our work concretizes this idea with the part-linking process depicted in Figure 1[1]. This figure shows two resources r_i and r_j where each resource is decomposed into a set of parts (p_{ix}, p_{jy}) and a global content measure (g_i, g_j). The two resources are then "centered" and a line is drawn between each pair of parts upholding a "sufficient" degree of similarity. Furthermore, a line is drawn between the global content measures if they uphold a "sufficient" degree of similarity. Finally, the semantic association assigned to $r_i \rightarrow r_j$ is a function of (1) the degree of global (g_i/g_j) match; (2) the common content required for parts to match; and (3) the *pattern* of the lines connecting matched parts. The third criteria is the innovation of the part-linking work and remains an open area for further exploration. The pattern will be some function of the number of parts matched, the space in between matching parts, the crossings of the lines drawn between the matched parts, or other features that can be extracted from the structure shown in Figure 1.

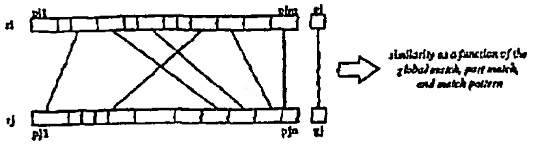

ri
pi1
pim
gi
rj
pj1
pjn
gj
similarity as a function of the global match, part match, and match pattern

Figure 1: Part Decomposition and Matching

3 The HTML-Linking Model

Html-linking captures the part-linking pattern between resources as a 6-place real-valued association vector $(\bar{A})$:

$$\begin{aligned} assoc(r_s, r_d, m_k) = \bar{A}_{sd} &= (c_{sd}, d_{sd}, u_{sd}, s_{sd}, f_{sd}, l_{sd}) \\ c_{sd}, d_{sd}, u_{sd}, s_{sd}, f_{sd}, l_{sd} &\in [0 \ldots 1] \end{aligned}$$

where r_s is the target resource, r_d is the resource we wish to associate with r_s, and $m_k \in [0 \ldots 1]$ is the minimum similarity required for parts from r_s and r_d to be considered matching. Given a target resource r_s, a specific content match threshold m_k, and an association context $\bar{A}$, exact html-linking discovers the set R_d of all resources in the network such that each resource $r_j \in R_d$ upholds "$assoc(r_s, r_j, m_k) = \bar{A}$."

To extend to *approximate matching*, a neighborhood relaxation *window* is introduced. Given two association vectors, $\bar{A}_{ij}$ and $\bar{A}_{xy}$, the *window* function evaluates to *true* if $\bar{A}_{ij}$ is in the $w_k \in [0 \ldots 1]$ sized window around $\bar{A}_{xy}$:

$$window(w_k, \bar{A}_{ij}, \bar{A}_{xy}) \in [true, false]$$

Using the *assoc* and *window* functions, html-linking can be defined as a service taking four inputs and finding all resources r_j in the network such that:

$$\begin{aligned} &\text{html-linking}(r_s, \bar{A}, m_k, w_k) \mapsto R_d \\ &\text{such that } \forall r_j \in R_d \\ &assoc(r_s, r_j, m_k) = \bar{A}_{sj} \text{ and} \\ &\qquad window(w_k, \bar{A}_{sj}, \bar{A}) = true \end{aligned} \tag{1}$$

A target request such as, "Find all resources similar to r_s in the same manner that r_d is similar to r_s," will generate an html-linking request with the target association context as $\bar{A} = assoc(r_s, r_d, m_k)$. The remainder of this section defines the constructs introduced in Equation 1 and demonstrates how html-linking performs a *specialized* part-linking search and match over HTML resources.

3.1 Basis for HTML Resource Decomposition

3.1.1 HTML Overview

HTML is a WWW standard [15] for annotating "raw" text documents with markers that add information to the chunk of text contained within each individual marker. The purpose of the markup commands are to attach coherent categories to clusters of text and separate the otherwise sequential document into logical elements. Figure 2 shows a portion of an HTML document – each tag consists of: a begin tag marker, the text to be interpreted in this tag environment, and an end tag marker. For html-linking purposes, the HTML tag set can be considered to consist of two classes of tags:

[1] The work in [1] followed a significantly different approach for applying this general "part decomposition and analysis" idea.

1. Sectioning and organization tags group chunks of text into logical elements. Examples of tags in this class are paragraph markers, list markers, line breaks, section headings, titles, and tables.
2. Extrinsic tags associate chunks of text with information external to the document itself. Examples of tags in this class are anchors, image references, and Java applets. Of particular interest is the "anchor" tag which associates a chunk of text in the document with any URL-addressable external information object.

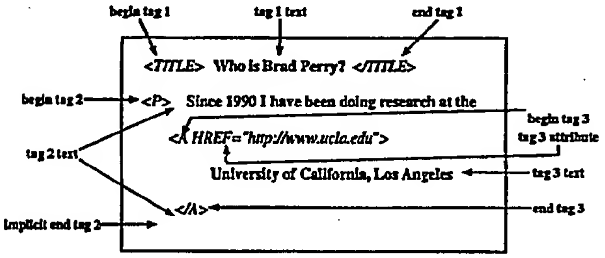

begin tag 1
tag 1 text
end tag 1
<TITLE> Who is Brad Perry? </TITLE>
begin tag 2
<P> Since 1990 I have been doing research at the
begin tag 3
tag 3 attribute
<A HREF="http://www.ucla.edu">
tag 2 text
University of California, Los Angeles
tag 3 text
</A>
end tag 3
implicit end tag 2

Figure 2: General Anatomy of an HTML Resource

We should be careful to identify the difference between our html-linking process, a value-added external information service, and the anchor tags internal to HTML resources. HTML anchors are *typeless* links – every link has the same appearance regardless of the media type at the other end of the link and regardless of the semantic association intended by the resource author. The accuracy of each link is constrained by the author's limited knowledge of what resources exist throughout the entire network. In addition, every user sees the exact same set of anchors in a document regardless f the profile and goals of the user's information processing session. Nevertheless, HTML anchors are an important metaphor for pointing resources to related media objects. The caution is that the "relation" should only and can only be taken to imply casual association. Html-linking, on the other hand, is a service *external* to the actual resources and based heavily on the typed semantic association that can be inferred among resources. HTML anchors provide casual and locally computed media associations in the web whereas value-added services (such as html-linking) provide heavily typed and far-reaching associations discovered amongst large collections of resources.

3.1.2 Part Decomposition

For HTML resources, parts are taken to be *contiguous, delineated sentence groups*. In order to identify parts, the document is searched for text sectioning and organization tags to use in extracting delineated sentence groups. In summary the following general decomposition was developed:

- Table cells are aggregated and each table is processed as a single sentence group (i.e., part).
- List elements are aggregated and each list is processed as a single sentence group.
- Traditional paragraphs are kept together and processed as single sentence groups.
- Titles and section headings are treated as independent parts.

A single level of this decomposition is performed; that is, nested sectioning and organization tags do not create new parts although they are used to properly collect the text into contiguous chunks. For example, a resource using two lists with tables inside the elements of one list and nested lists in the other would result in two parts – those corresponding to the top-level lists themselves. Using the data sets described in Section 4, it was observed that, using this decomposition parser, the number of parts in the HTML resources ranged from 2 to 40 with an average of 14; and the number of terms in each part ranged from 4 to 78 with an average of 28.

As a final step, each part generated by the above "decomposition process" is annotated with the top level HTML tag that caused the part to be extracted. In other words, each part is attached one of the following labels: *title, heading, table, list*, or *paragraph*. This labeling does not provide the deep semantic structuring performed in more closed and domain-specific studies; however it does provide a portable structuring that can be applied to HTML documents in general. The overall decomposition performed on HTML resources is summarized as follows:

$$\begin{aligned} decomp(r_i) \;&=\; \{(p_{ik}, l_k) \mid p_{ik} \text{ is a delineated text chunk } \} \\ &\quad\; p_{ik} = \text{terms in the } kth \text{ part of resource } r_i \\ &\quad\; l_k \in [title,\ heading,\ table,\ list,\ paragraph] \end{aligned}$$

3.1.3 Part Similarity

When performing part-linking, the parts from each resource are compared to see if they match with "sufficient" similarity. Thus, a similarity function is assumed that, given two parts p_i and p_j, computes a number in the range $[0\ldots1]$. This function returns 1.0 if the parts are an exact match and 0.0 if they have no content in common. The emphasis of this work is not to develop the "correct" function for comparing resource parts but, rather, to allow any part similarity function to be plugged into the part-linking process. Given that two parts are simply a set of terms and a structure label, the following similarity function was used in the experiments with html-linking:

- If the parts have different labels, then their similarity is 0.0.
- If the parts have the same label, then their similarity is computed as the ratio of the number of terms they have in common to the total number of terms they share. The reader is referred to [13] for a general review of such standard term-based techniques for comparing textual passages.

The html-linking model incorporates a control parameter that sets the minimum similarity parts must exhibit to align and contribute to the part-linking pattern.

3.2 Basis for HTML Resource Linking

3.2.1 Features for Match Patterns

Recall that the emphasis behind part-linking is to "line up" two resources, "draw" lines between matching parts, and then "analyze" the resulting patterns, as shown in Figure 1. Html-linking proceeds to characterize the pattern between any pair of resources in a real-valued vector. One tunable control parameter and four features are defined for capturing a part pattern from an input HTML resource to a specific destination resource. Figure 3 shows some resource pairs and the match patterns they exhibit.

1. *minimum match tolerance*, $m = m_k \in [0 \ldots 1]$, controls the minimum similarity parts must exhibit before declared as matching. Referring to Figure 3, parts with a line drawn between them are parts that match with a threshold greater than this value.

2. *comprehension*, $c \in [0 \ldots 1]$, determines the coverage of the query resource. It is the number of parts (as a percentage of the total parts) from the query-resource that participate in the established pattern.

3. *diversity*, $d \in [0 \ldots 1]$, determines the coverage of the destination resource. It is the number of parts (as a percentage of the total parts) from the destination-resource that participate in the established pattern.

4. *unity*, $u \in [0 \ldots 1]$, determines the unity of the matched parts in the query-resource. To compute the unity, count the number of parts in the query-resource between the lowest and highest indexed matched parts – call this count the *pack_len*. Then count the number of matched parts in the query-resource – call this count the *match_len*. The value

   $$match\_len/pack\_len$$

   is the "packing factor", or unity, of the pattern.

5. *scatter*, $s \in [0 \ldots 1]$, determines the unity of the matched parts in the destination-resource. As with the *unity* feature, count the *pack_len* and *match_len* of the parts in the destination-resource. The value

   $$match\_len/pack\_len$$

   is the destination packing factor of, or scatter of matching parts in, the pattern.

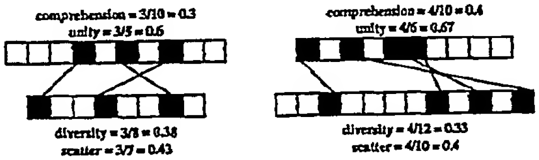


Figure 3: Part Match Pattern Examples

The vector *(c, d, u, s)* captures the cohesion of the locally and structurally confined content matches spread across two HTML resources.

3.2.2 Global Resource Features

The extrinsic HTML tags are used to create a global behavior to track among resources. The "anchor tendencies" of an HTML resource define two tunable global content features:

1. *fanout*, $f \in [0 \ldots 1]$, determines the amount of anchoring a destination resource contains. First, count the number of terms in the destination resource – call this count *term_total*. Then count the number of terms in the resource that appear within anchor markup tags – call this count *anchor_total*. The value

   $$anchor\_total/term\_total$$

   is the anchoring, or fanout, of the destination resource.

2. *liveliness*, $l \in [0 \ldots 1]$, determines the amount of multimedia anchoring a destination resource contains. First, count the number of anchors in the destination resource – call this count *anchors*. Then count the number of anchors that point to either image, sound, or movie resources (i.e., multimedia resources) – call this count the *action_anchors*. The value

   $$action\_anchors/anchors$$

   is the liveliness of the destination resource.

The vector (f, l) captures a global anchoring behavior exhibited by a resource. The behavior creates a spectrum for web resources spanning from those relatively "content self-contained" to those relatively "content referential". Of those resources exhibiting referential tendencies, a spectrum is established between those resources incorporating high degrees of multimedia to those referring to inanimate external resources. A search for similar resources can constrain this vector to the specific hypermedia behaviors to be discovered. As identified by the AltaVista WWW search engine [3], querying the multimedia aspects of web resources is an attractive feature that users frequently request and employ when available.

3.2.3 Summary: HTML Association Contexts

The part-linking correspondence between HTML resources is captured in a real-valued *association vector*. The association from resource r_s to resource r_d is captured as:

$$\begin{aligned} assoc(r_s, r_d, m_k) &= \bar{A}_{sd} = (c_{sd}, d_{sd}, u_{sd}, s_{sd}, f_{sd}, l_{sd}) \\ c_{sd} &= \text{comprehension of } r_s \in [0 \ldots 1] \\ d_{sd} &= \text{diversity of } r_d \in [0 \ldots 1] \\ u_{sd} &= \text{unity of } r_s \in [0 \ldots 1] \\ s_{sd} &= \text{scatter of } r_d \in [0 \ldots 1] \\ f_{sd} &= \text{fanout of } r_d \in [0 \ldots 1] \\ l_{sd} &= \text{liveliness of } r_d \in [0 \ldots 1] \end{aligned}$$

where $m_k \in [0 \ldots 1]$ specifies the minimum similarity required for two parts to be considered a match in the association of r_s with r_d. Essentially, the context vector between two resources can be viewed as expressing the *type of association* between the resources. Therefore, pairs of resources that induce the same (or similar) context vector are pairs that can be considered as "associated in the same type."

3.3 Neighborhood Search Model

Given a known resource r_s, a target vector $\bar{A}$ and a match threshold m_k defines an exact search for

all resources r_d such that $assoc(r_s, r_d, m_k) = \bar{A}$

The *neighborhood search model* is introduced to resolve the following problem with the exact model: if

$$assoc(r_s, r_d, m_k) = \bar{A}_{sd} \text{ and } \bar{A}_{sd} \neq \bar{A}$$

then can $\bar{A}_{sd}$ be sufficiently close to $\bar{A}$ to still qualify r_d as a positive match? A window, or neighborhood, around $\bar{A}$ is introduced for identifying this style of approximately matching associations. Assume the existence of a window size $w_k \in [0 \ldots 1]$, an association context $\bar{A}$, an origin r_s, and a resource r_d such that $assoc(r_s, r_d, m_k) = \bar{A}_{sd}$. Then, r_d is a positive association to r_s in window w_k if:

$$window(w_k, \bar{A}_{sd}, \bar{A}) = true$$

The *window* function operates on two association vectors, $\bar{A}_{ij}$ and $\bar{A}_{xy}$ and returns *true* if $\bar{A}_{ij}$ is in the w_k sized window around $\bar{A}_{xy}$.

$$\begin{aligned}
&\text{if } \bar{A}_{ij} = (c_{ij}, d_{ij}, u_{ij}, s_{ij}, f_{ij}, l_{ij}) \text{ and}\\
&\quad \bar{A}_{xy} = (c_{xy}, d_{xy}, u_{xy}, s_{xy}, f_{xy}, l_{xy})\\
&\text{then } window(w_k, \bar{A}_{ij}, \bar{A}_{xy}) = true \iff\\
&\quad c_{ij} \in [c_{xy} - w_k/2 \ldots c_{xy} + w_k/2] \text{ and}\\
&\quad d_{ij} \in [d_{xy} - w_k/2 \ldots d_{xy} + w_k/2] \text{ and}\\
&\quad u_{ij} \in [u_{xy} - w_k/2 \ldots u_{xy} + w_k/2] \text{ and}\\
&\quad s_{ij} \in [s_{xy} - w_k/2 \ldots s_{xy} + w_k/2] \text{ and}\\
&\quad f_{ij} \in [f_{xy} - w_k/2 \ldots f_{xy} + w_k/2] \text{ and}\\
&\quad l_{ij} \in [l_{xy} - w_k/2 \ldots l_{xy} + w_k/2]
\end{aligned}$$

In other words, resource r_d is in the w_k window of association to resource r_s if all elements in $\bar{A}_{sd}$ are within $w_k/2$ of the corresponding elements in target vector $\bar{A}$.

Each request, in the neighborhood search process, begins with an originating source r_s and retrieves a *neighborhood* of associated destination sources $R_d = \{r_d\}$. In other words, the search request "html-linking$(r_s, \bar{A}, m_k, w_k)$" is submitted for matching in the information network. The following dichotomy exists between the inputs to the html-linking request:

- Inputs r_s and $\bar{A}$ determine the *semantics*, or style of association, to search for – r_s is the example resource upon which to base the association and $\bar{A}$ is the pattern, or link type, with respect to r_s to retrieve. As a result, changing any part of r_s or $\bar{A}$ will change the underlying meaning of the request.
- Inputs m_k and w_k determine the *neighborhood* and quality of the search. Changing either m_k or w_k will not change the meaning of the result set – it will just change the size and relevance of the items qualified as matching the context $(r_s, \bar{A})$.

Decreasing the value of m_k or increasing the value of w_k will increase the size of R_d. We would like m_k to get as small as possible without allowing irrelevant parts to positively match during the association computation. In addition, we would like w_k to get as big as possible without allowing irrelevant patterns to positively match during the association computation. This transition from relevant to irrelevant relaxations, and the m_k and w_k values it entails, is demonstrated in the next section.

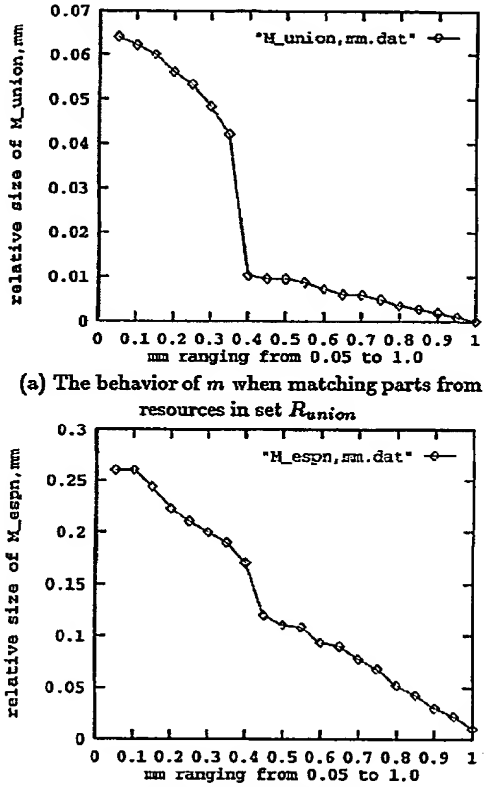
4 Experimental Observations

4.1 Data Sets

To experiment with the behavior of html-linking, five data sets were gathered from live WWW sources, as shown in Table 1. The last row, set R_{union} represents the union of all five sets. The resources gathered from these sources were inserted into an html-linking index [11] and the experiments described in the next sections were performed.

4.2 Minimum Content Match

This experiment observed the behavior of the minimum content match control (parameter $m = m_k \in [0 \ldots 1]$) in the html-linking model. In the experiments, all pairs of resources in the data sets were examined, one pair at a time, under varying thresholds for m. At each value $m = m_k$, it was computed what percentage of the possible resource pairs exhibited at least one part matching above threshold m_k. Figure 4 shows the graphs for resource sets R_{union} and R_{espn}, the graphs for the other 4 data sets showed similar behavior.



(a) The behavior of m when matching parts from resources in set R_{union}

(b) The behavior of m when matching parts from resources in set R_{espn}

Figure 4: Content Match Plots

The goal of this experiment was to identify if, and where, decreasing the m parameter tends to introduce irrelevant content matches. The "point of irrelevance" is defined to be the point in Figure 4 where the percentage of resources with "matchable" parts demonstrates a noticeable spike. The intuition being that such a spike represents the point where the content is matching with such a minimal required threshold that all parts begin to look like matching chunks of text. The interesting fact is that the spike for parameter m occurred very close in both the *union* and *espn* data sets (as it also did in the *nando*, *ucla*, *stan*, and *cnn* data sets that are not shown here).

> When parameter $m \approx 0.4$, an experimentally observed break in relaxation occurs where it is expected that matching content ceases to hold relevance to the intended match context.

Therefore a range of $[0.4 \ldots 1.0]$ can be used as the exper-

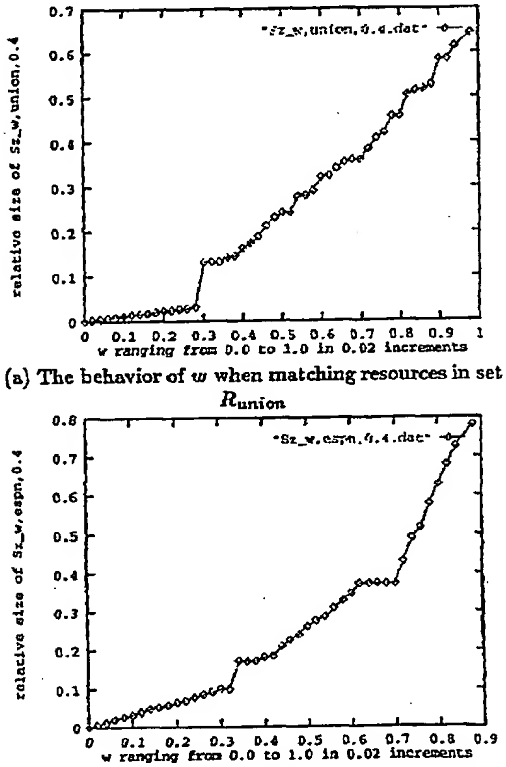
Name	Domain	Net Location	# of Resources Gathered
R_{cnn}	CNN online news	www.cnn.com	4303
R_{espn}	ESPN online sports server	espnet.sportszone.com	2918
R_{nando}	Nando integrated news/sports server	www.nando.net	5845
R_{ucla}	UCLA computer science dept.	www.cs.ucla.edu	1923
R_{stan}	Stanford computer science dept.	www.cs.stanford.edu	2349
R_{union}			17338

Table 1: Case Study Resource Gatherings

imentally observed "useful" range of values for control parameter m.

4.3 Maximum Pattern Window

This experiment observed the behavior of the maximum pattern window control (parameter $w = w_k \in [0 \ldots 1]$) in the html-linking model. In the experiments, all pairs of resources with at least one part matching above $m = 0.4$ were examined, one pair at a time, under varying thresholds for w. At each value $w = w_k$, we computed the average number of resources (as a percentage of all possible resource pairs) exhibiting patterns that match in a relaxation window of size w_k. Figure 5 shows the graphs for resource sets R_{union} and R_{espn}, the graphs for the other 4 data sets showed similar behavior.



(a) The behavior of w when matching resources in set R_{union}

(b) The behavior of w when matching resources in set R_{espn}

Figure 5: Pattern Window Plots

The goal of this experiment was to identify if, and where, increasing the w parameter tends to introduce irrelevant resource patterns into a match context. The "point of irrelevance" is defined to be the point in Figure 5 where the number of resources with "matchable" parts demonstrates a noticeable spike. The intuition being that such a spike represents the point where the match context is so relaxed that all resources tend to match the desired context. The interesting fact is that the spike for parameter w occurred very close in both the *union* and *espn* data sets (as it also did in the *nando*, *ucla*, *stan*, and *cnn* data sets that are not shown here).

> When parameter $w \approx 0.3$, an experimentally observed break in relaxation occurs where it is expected that matching patterns cease to hold relevance to the intended match context.

Therefore a range of $[0.0 \ldots 0.3]$ can be used as the experimentally observed "useful" range of values for control parameter w.

4.4 Experimental Summary

This section presented a "snapshot" of the type of behavior observed when matching html-linking association contexts under varying content and window relaxations. In the case of content relaxations, it was seen that a range of $[0.4 \ldots 1.0]$ for the m control parameter tended to yield useful results over various samples of resources. In the case of window relaxations, it was seen that a range of $[0.0 \ldots 0.3]$ for the w control parameter tended to yield useful results over various samples of resources. These experiments give a flavor for the general *observed behavior* of html-linking. In [11] these observations are coupled with an analytical model of the effects the association vector $\bar{A}$ under varying relaxations. The coupling results in an integrated control model, for adjusting the m and w parameters, that predicts the quality of the results expected in the html-linking result sets. The intent in this paper is to introduce html-linking and demonstrate that it tends to show correlated behavior when evaluated over diverse resource collections.

5 Summary and Future Work

In this paper we introduced the *html-linking* model for computing the content association between two HTML resources. The association is based on the following three criteria:

1. Decomposing resources into coherent units, or passages.
2. Finding "content matching" passages between two resources.
3. Computing the "structural match pattern" exhibited by the content matching passages between two resources. This structural match pattern describes the type of association the resources possess.

Our ongoing work involves expanding the methods used in the first and third criteria listed above, as summarized below.

5.1 Res urce Decomposition

Our current *resource decomposition* strategy is dependent on the existence of HTML tags in the resources. These tags are interpreted as structural markers and a resource is decomposed into passages, or parts, based on the placement of these tags throughout the document. This decomposition strategy has worked well for decomposing HTML documents encountered on the web, but is an area for much improvement in our implementation. There are two interesting extensions to this decomposition strategy that we are investigating.

First, the *semantic consistency* of the parts we extract can be solidified. In work such as that done in [9], techniques are developed for segmenting textual streams into semantically consistent passages. That is, controlled *natural language understanding* techniques are coupled with structural organization cues to create "semantic segmentations" of textual objects into consistent passages. The segmentation is termed "semantic" because it attempts to keep sentences in the same passages when they are discussing the same theme. Therefore, the segmentation mediates between to decomposition criteria: (1) the semantic, or theme-driven, delineation of text; and (2) the actual decomposition of the text into sentences, paragraphs, and other structural units. Our current decomposition uses only the second criteria (as identified by the HTML organization tags). It is anticipated that using theme consistency to aid the decision to segregate or aggregate text will lead to more coherent and consistent passage decompositions.

Second, our focus has been on strictly the textual aspects of documents encountered on the web. Yet, the web is a diverse multimedia information space and documents are often aggregations of text, images, video, and various executable contents. Currently we are ignoring these non-textual components in our document decomposition and analysis. Our future work involves extending the passage decomposition of HTML resources to include features and other aspects of the non-textual elements interspersed in the documents.

5.2 Match Patterns

In this paper, we presented a 6-dimensional feature vector used to capture the association between resources. We are actively exploring various classes of additional features that can be used to describe and match the part pattern exhibited between resources in large and unconstrained collections [11]. These "new features" fall into two classes. First, there are the features that build from our existing decomposition strategy and simply add additional match pattern descriptors. In [2], for example, a similar part-linking approach is used where the number of *cross-links* exhibited by the lines between matching parts (refer back to Figure 1) are counted and used to extract resource similarities. Second, there are the features that manifest from the more advanced decompositions discussed above. In the example of using themes to semantically drive the text decomposition, it is the case that the semantic distance between contiguous passages is not always the same. Thus, when we are aligning two resources, as in Figure 1, it may be the case that the parts can be placed on a linear scale representing there relative semantic similarity to one another. This scale could provide additional information when computing the cohesiveness of matching parts between two documents. In the example of incorporating descriptors of images and other non-textual entities in the documents, the association feature space would need to be extended to properly match, align, and describe the association between these elements of the document decomposition. In either case, it is clear that our current 6-dimensional vector has provided promising results but can be expanded to incorporate additional general and domain-specific features of the association between two documents.

5.3 Conclusion

The part-linking basis, upon which html-linking is built, identifies a promising technique for comparing resources based on the content and structural associations exhibited among their parts. We have developed and demonstrated html-linking as an external information service that adds value to sets of information collections found on the web. Html-linking can be used as the resource comparison method at the center both link discovery and link annotation services. The content locality and organizational structure sensitivity of part-linking make it a promising technique to further explore as a vehicle for identifying semantic resource associations on the web.

References

[1] J. Allan. *Automatic Hypertext Construction*. Ph.D. dissertation, Cornell University, 1995.

[2] J. Allan. Automatic hypertext link typing. In *Hypertext-96: Seventh ACM Conference on Hypertext*, 1996.

[3] AltaVista. *AltaVista Search: Main Page*. http://www.altavista.digital.com/, 1995.

[4] R. Armstrong, D. Freitag, T. Joachims, and T. Mitchell. Webwatcher: A learning apprentice for the world wide web. In *1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, 1995.

[5] M. Bernstein. An apprentice that discovers hypertext links. In *Hypertext: Concepts, Systems, and Applications: European Conference on Hypertext*, 1990.

[6] Vannevar Bush. As we may think. *Atlantic Monthly*, July 1945.

[7] F. Can. Incremental clustering for dynamic information processing. *ACM Transactions on Information Systems*, 11(2), April 1993.

[8] C. Cleary and R. Bareiss. Practical methods for automatically generating typed links. In *Hypertext-96: 7th ACM Conference on Hypertext*, 1996.

[9] M. Hearst. Multi-paragraph segmentation of expository text. In *ACL-94*, 1994.

[10] L. Page. *BackRub WebCrawler*. http://backrub.stanford.edu/, 1996.

[11] B. Perry. *Dynamically Discovering Similar Resources in Large-scale Information Networks*. Ph.D. dissertation, University of California, Los Angeles, 1997.

[12] A. Salminen, J. Tague-Sutcliffe, and C. McClellan. From text to hypertext by indexing. *ACM Transactions on Information Systems*, 13(1), January 1995.

[13] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer.* Addison-Wesley Publishing Company, 1989.

[14] W3C. Relationships in HTML links. Technical Report http://www.w3.org/pub/WWW/MarkUp/Relationships, World Wide Web Consortium, May 1996.

[15] W3C. *World Wide Web Consortium.* http://www.w3.org/, 1996.

[16] R. Weiss, B. Velez, M. Sheldon, C. Namprempre, P. Szilagyi, A. Duda, and D. Gifford. Hypursuit: A hierarchical network search engine that exploits content-link hypertext clustering. In *Hypertext-96: 7th ACM Conference on Hypertext*, 1996.

[17] T.W. Yan, M. Jacobsen, H. Garcia-Molina, and U. Dayal. From user access patterns to dynamic hypertext linking. Stanford University: Working Papers, 1996.

[18] L. De Young. Links considered harmful. In *Hypertext: Concepts, Systems, and Applications: European Conference on Hypertext*, 1990.